How to Secure Your Software Supply Chain and Speed-Up DFIR with Hashlookup

the harsh reality of the software supply chain

CIRCL
Computer Incident
Response Center
Luxembourg

Alexandre Dulaunoy
Jean-Louis Huynen

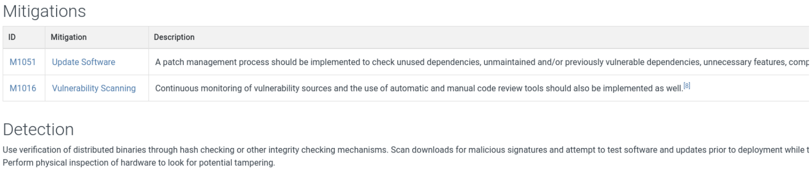
TLP:WHITE

info@circl.lu / hashlookup.io

FIRST - Dublin - 30 June 2022

ATT&CK Technique: Supply Chain Compromise (T1195)

- *Adversaries may manipulate products or product delivery mechanisms prior to receipt by a final consumer for the purpose of data or system compromise.*
- **Use verification of distributed binaries through hash checking**. But is this easy? Where can you find those hashes?



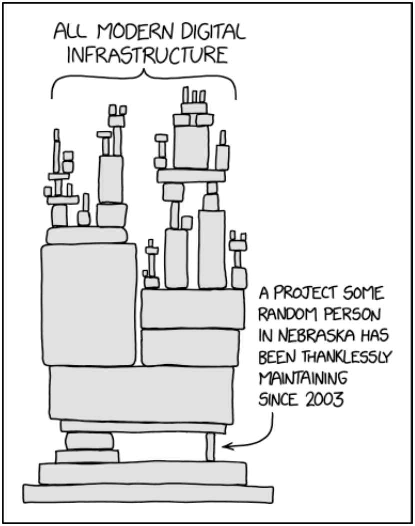
Mitigations

ID	Mitigation	Description
M1051	Update Software	A patch management process should be implemented to check unused dependencies, unmaintained and/or previously vulnerable dependencies, unnecessary features, comp
M1016	Vulnerability Scanning	Continuous monitoring of vulnerability sources and the use of automatic and manual code review tools should also be implemented as well.[8]

Detection

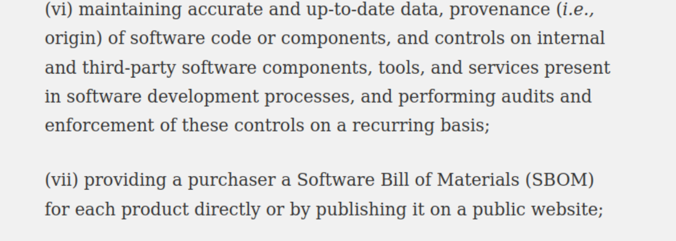
Use verification of distributed binaries through hash checking or other integrity checking mechanisms. Scan downloads for malicious signatures and attempt to test software and updates prior to deployment while t
Perform physical inspection of hardware to look for potential tampering.

Do you know about this little binary used everywhere?



ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

US - Executive Order 14028 of May 12, 2021



(vi) maintaining accurate and up-to-date data, provenance (*i.e.,* origin) of software code or components, and controls on internal and third-party software components, tools, and services present in software development processes, and performing audits and enforcement of these controls on a recurring basis;

(vii) providing a purchaser a Software Bill of Materials (SBOM) for each product directly or by publishing it on a public website;

[1]

- SolarWinds was just a trigger,
- Havex (ICS distribution), Kingslayer (repackaging signed binaries), CCleaner (build environment), NetSarang (Backdooring a Windows Updater), ASUS (custom updater), software repositories (npm, PyPI)...

[1] `https://www.federalregister.gov/documents/2021/05/17/2021-10460/improving-the-nations-cybersecurity`

Starting digital forensic investigation on a recent acquisition

- A single disk acquisition of a desktop or server operating system contains at least 150K files,
- Large portion of directories and files are not analysed due to a **lack of time**,
- Finding legitimate versus attacker-installed files can be difficult if the timeline is incorrect,
- Many legacy tools are used by attackers and mixed with custom binaries.

Known file filters - DFIR issues

- **State of current NIST NSRL**[2] databases and other known file filters (KFF)
- A lack of Operating Systems / Software available (e.g. OSX?, Linux distributions)
- nsrllookup.com / nsrlsrv use their own protocol, no ReST API
- nsrlsrv[3] only supports MD5s
- Many **sources are difficult to use** (e.g. NSRL ISOs/SQLite), **ill-maintained**, **outdated** or **expensive**,
- MISP integration (malicious hashes versus known hashes).

[2] https://www.nist.gov/itl/ssd/software-quality-group/national-software-reference-library-nsrl

[3] https://rjhansen.github.io/nsrlsvr/

Indexing all published software?

- **Regular updates of Linux distributions** including security updates on multiple architectures,
- 800+ software releases per hour on GitHub
- Bundling of software in **snap** images, **flatpak**, **AppImage**, etc.
- **Continuous release** of security updates
- Microsoft Windows and Apple custom software distribution schemes.

Known file filters - improvements required

- A need for a **public, open and easy** to use API for all sources (NSRL is not alone)
- A **global, public instance of all known sources**,
- A common ReST API normalises the access to several datasources
- Available for MD5, and SHA1 (and more)
- Includes fuzzy hashes
- Includes additional datapoints available by **combining a set of datasources**

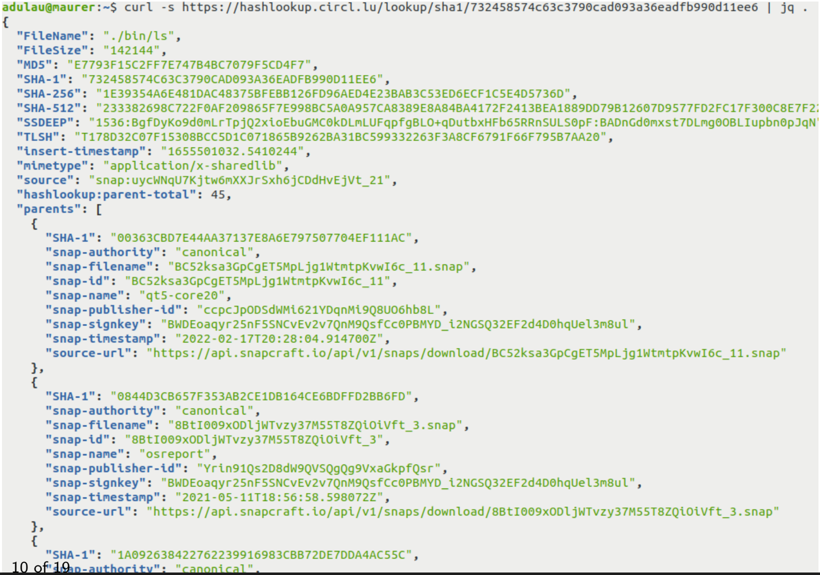
CIRCL hashlookup public service

- https://hashlookup.circl.lu/[4] - **OpenAPI** Swagger[5]
- NIST NSRL - **all RDS hash sets** including current, modern, android, iOS and legacy sets
- Ubuntu package distribution
- CentOS core OS distribution
- Fedora project EPEL repository
- CDNjs repository
- Kali linux package distribution, OpenSUSE distribution and **more**
- **If you find it in a lot of trusted places, you may find that it's reasonable to trust it**.

[4] https://hashlookup.circl.lu/

[5] https://hashlookup.circl.lu/swagger.json

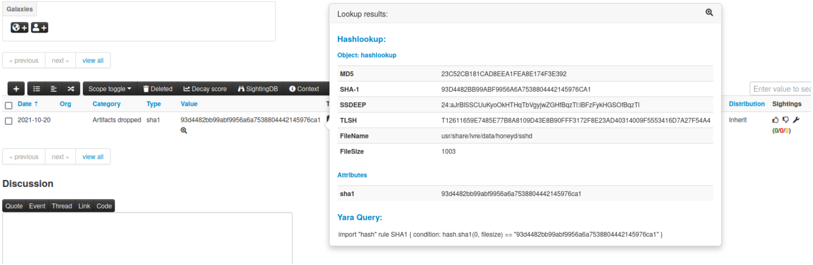
hashlookup.circl.lu API example



```
adulau@maurer:~$ curl -s https://hashlookup.circl.lu/lookup/sha1/732458574c63c3790cad093a36eadfb990d11ee6 | jq .
{
  "FileName": "./bin/ls",
  "FileSize": "142144",
  "MD5": "E7793F15C2FF7E747B4BC7079F5CD4F7",
  "SHA-1": "732458574C63C3790CAD093A36EADFB990D11EE6",
  "SHA-256": "1E39354A6E481DAC48375BFEBB126FD96AED4E23BAB3C53ED6ECF1C5E4D5736D",
  "SHA-512": "233382698C722F0AF209865F7E998BC5A0A957CA8389E8A84BA4172F2413BEA1889DD79B12607D9577FD2FC17F300C8E7F2
  "SSDEEP": "1536:BgfDyKo9d0mLrTpjQ2xioEbuGMC0kDLmLUFqpfgBLO+qDutbxHFb65RRnSULS0pF:BADnGd0mxst7DLmg0OBLIupbn0pJqN
  "TLSH": "T178D32C07F15308BCC5D1C071865B9262BA31BC599332263F3A8CF6791F66F795B7AA20",
  "insert-timestamp": "1655501032.5410244",
  "mimetype": "application/x-sharedlib",
  "source": "snap:uycWNqU7Kjtw6mXXJrSxh6jCDdHvEjVt_21",
  "hashlookup:parent-total": 45,
  "parents": [
    {
      "SHA-1": "00363CBD7E44AA37137E8A6E797507704EF111AC",
      "snap-authority": "canonical",
      "snap-filename": "BC52ksa3GpCgET5MpLjg1WtmtpKvwI6c_11.snap",
      "snap-id": "BC52ksa3GpCgET5MpLjg1WtmtpKvwI6c_11",
      "snap-name": "qt5-core20",
      "snap-publisher-id": "ccpcJpODSdWMi621YDqnMi9Q8UO6hb8L",
      "snap-signkey": "BWDEoaqyr25nF5SNCvEv2v7QnM9QsfCc0PBMYD_i2NGSQ32EF2d4D0hqUel3m8ul",
      "snap-timestamp": "2022-02-17T20:28:04.914700Z",
      "source-url": "https://api.snapcraft.io/api/v1/snaps/download/BC52ksa3GpCgET5MpLjg1WtmtpKvwI6c_11.snap"
    },
    {
      "SHA-1": "0844D3CB657F353AB2CE1DB164CE6BDFFD2BB6FD",
      "snap-authority": "canonical",
      "snap-filename": "8BtI009xODljWTvzy37M55T8ZQiOiVft_3.snap",
      "snap-id": "8BtI009xODljWTvzy37M55T8ZQiOiVft_3",
      "snap-name": "osreport",
      "snap-publisher-id": "Yrin91Qs2D8dW9QVSQgQg9VxaGkpfQsr",
      "snap-signkey": "BWDEoaqyr25nF5SNCvEv2v7QnM9QsfCc0PBMYD_i2NGSQ32EF2d4D0hqUel3m8ul",
      "snap-timestamp": "2021-05-11T18:56:58.598072Z",
      "source-url": "https://api.snapcraft.io/api/v1/snaps/download/8BtI009xODljWTvzy37M55T8ZQiOiVft_3.snap"
    },
    {
      "SHA-1": "1A0926384227622399169 83CBB72DE7DDA4AC55C",
      "snap-authority": "canonical",
```

hashlookup MISP module

- A hover and expansion module[6] to quickly check if a hash is part of the known files of hashlookup:



Date	Org	Category	Type	Value	Distribution	Sightings
2021-10-20		Artifacts dropped	sha1	93d4482bb99abf9956a6a7538804442145976ca1	Inherit	(0/0/0)

Lookup results:

Hashlookup:

Object: hashlookup

MD5	23C52CB181CAD8EEA1FEA8E174F3E392
SHA-1	93D4482BB99ABF9956A6A7538804442145976CA1
SSDEEP	24:aJrBlSSCUuKyoOkHTHqTbVgyjwZGHfBqzTl:iBFzFykHGSOfBqzTl
TLSH	T12611659E7485E77B8A8109D43E8B90FFF3172F8E23AD40314009F5553416D7A27F54A4
FileName	usr/share/ivre/data/honeyd/sshd
FileSize	1003

Attributes

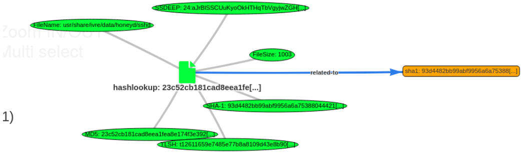
sha1	93d4482bb99abf9956a6a7538804442145976ca1

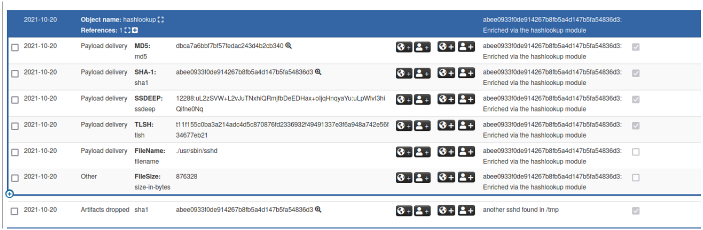
Yara Query:

```
import "hash" rule SHA1 { condition: hash.sha1(0, filesize) == "93d4482bb99abf9956a6a7538804442145976ca1" }
```

[6] https://misp.github.io/misp-modules/expansion/#hashlookup

hashlookup MISP module - import


SSDEEP: 24:aJrBISSCUuKyoOkHTHqTbVgyjwZGH[...]
FileName: usr/share/ivre/data/honeyd/sshd
Zoom IN/OUT
Multi select
FileSize: 1003
related-to
sha1: 93d4482bb99abf9956a6a75388[...]
hashlookup: 23c52cb181cad8eea1fe[...]
1)
SHA-1: 93d4482bb99abf9956a6a75388044421[...]
MD5: 23c52cb181cad8eea1fea8e174f3e392[...]
TLSH: t12611659e7485e77b8a8109d43e8b90[...]



	2021-10-20	**Object name:** hashlookup **References:** 1			abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	
□	2021-10-20	Payload delivery	**MD5:** md5	dbca7a6bbf7bf57fedac243d4b2cb340	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**SHA-1:** sha1	abee0933f0de914267b8fb5a4d147b5fa54836d3	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**SSDEEP:** ssdeep	12288:uL2zSVW+L2vJuTNxhiQRmjfbDeEDHax+oljqHnqyaYu:uLpWlvl3hi Qifne0Nq	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**TLSH:** tlsh	t11f155c0ba3a214adc4d5c870876fd2336932f49491337e3f6a948a742e56f 34677eb21	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**FileName:** filename	./usr/sbin/sshd	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	□
□	2021-10-20	Other	**FileSize:** size-in-bytes	876328	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	□
□	2021-10-20	Artifacts dropped	sha1	abee0933f0de914267b8fb5a4d147b5fa54836d3	another sshd found in /tmp	☑

hashlookup - offline lookup with Bloom filters

- DFIR requires **fast-lookup** and **offline** (for privacy and confidentiality reasons).
- hashlookup provides a weekly Bloom filter dump[7] for this purpose (see rationale here[8]),
- Bloom filter can be loaded in tools such as hashlookup-forensic-analyser[9], hashlookup-gui[10], and many others.

[7] `https://cra.circl.lu/hashlookup/hashlookup-full.bloom`

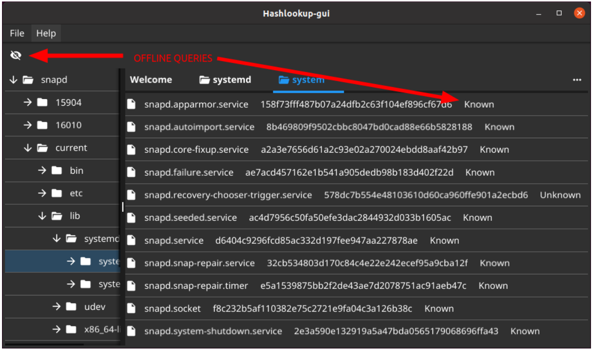
[8] `https://tinyurl.com/hashlookup-bloom`

[9] `https://www.github.com/hashlookup/hashlookup-forensic-analyser`

[10] `https://www.github.com/hashlookup/hashlookup-gui`

hashlookup-gui - offline lookups with Bloom filters

- hashlookup-gui[11] a multi-platform Graphical User Interface for querying hashlookup services.



snapd.apparmor.service	158f73fff487b07a24dfb2c63f104ef896cf67d6	Known
snapd.autoimport.service	8b469809f9502cbbc8047bd0cad88e66b5828188	Known
snapd.core-fixup.service	a2a3e7656d61a2c93e02a270024ebdd8aaf42b97	Known
snapd.failure.service	ae7acd457162e1b541a905dedb98b183d402f22d	Known
snapd.recovery-chooser-trigger.service	578dc7b554e48103610d60ca960ffe901a2ecbd6	Unknown
snapd.seeded.service	ac4d7956c50fa50efe3dac2844932d033b1605ac	Known
snapd.service	d6404c9296fcd85ac332d197fee947aa227878ae	Known
snapd.snap-repair.service	32cb534803d170c84c4e22e242ecef95a9cba12f	Known
snapd.snap-repair.timer	e5a1539875bb2f2de43ae7d2078751ac91aeb47c	Known
snapd.socket	f8c232b5af110382e75c2721e9fa04c3a126b38c	Known
snapd.system-shutdown.service	2e3a590e132919a5a47bda0565179068696ffa43	Known

[11] https://github.com/hashlookup/hashlookup-gui

hashlookup-forensic-analyser

- Analyse[12] **a forensic target** to find and report files, which were found or not found, from the hashlookup public service or the Bloom filter from CIRCL's hashlookup.
- Lookup **live processes** on Linux (using /proc) to discover unknown processes.
- Generate machine-readable reports for forensic triage.

[12] `https://github.com/hashlookup/hashlookup-forensic-analyser`

What's the future for the adversaries?

- We are still at **basic supply chain attacks** compared to Ken Thompson's paper on "Reflections on Trusting Trust" [13] (1984),
- The increased sources of distribution channels (software repackaged in packages - **hiding the mess**)
- SolarWinds attacks are just **the tip of iceberg** when it comes to the security state of the software supply chain
- Software reuse is finally here but the risks of libraries dependencies are increasing.

[13] https://www.cs.cmu.edu/~rdriley/487/papers/Thompson_1984_ReflectionsonTrustingTrust.pdf

What can I do?

- Require your supplier to provide a **software bill of materials (SBOM)** for each software release
- **Exercise your incident response procedure** and most importantly review your capability to baseline the origin of the software installed
- **Verify the claims** of your software vendors/suppliers (e.g. zero dependencies)
- Acquire internal capabilities to **verify software release integrity**

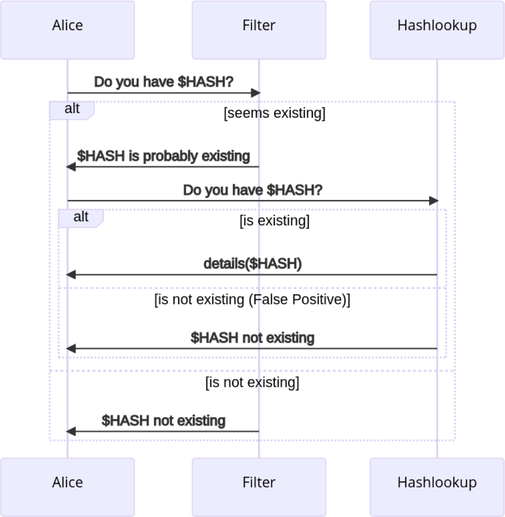
hashlookup.io future

- **Additional sources** of software publishers will be added on a regular basis
- Improving Bloom filters per type and categories of software
- Add an **API for known software publishers** to submit their hashes into hashlookup
- It's an open source project, so feel free to **contribute**

Contact

- `https://hashlookup.io/`
- `https://circl.lu/services/hashlookup/`
- Twitter: @adulau @circl_lu

What's up with Bloom filters? and API lookup?


Alice
Filter
Hashlookup
Do you have $HASH?
alt
[seems existing]
$HASH is probably existing
Do you have $HASH?
alt
[is existing]
details($HASH)
[is not existing (False Positive)]
$HASH not existing
[is not existing]
$HASH not existing
Alice
Filter
Hashlookup